Remarks/Arguments

A. Pending Claims

Claims 1-31 and 78 are pending. Claims 1, 2, 6, 21, 22, 26, 27, 31, and 78 have been amended.

B. Claim Objections

The Examiner objected to claims 1, 21, 26, and 31 because "Claim 1 has the acronym 'FSO' in the body of the claim and does not have what 'FSO' stands for in line 10 of the claim." The Examiner conducted a telephonic interview on October 28, 2004 with Chris Thompson, an associate of the undersigned, concerning these objections. The Examiner indicated that for each independent claim, the phrase "Financial Service Organization (FSO)" should appear in the body of the claim the first time it is used. The acronym "FSO" may be used by itself in the rest of the claim and in any claims dependent thereon. Applicant sincerely appreciates the Examiner's assistance in clarifying the objection. The claims have been amended accordingly. Applicant respectfully requests removal of these objections.

C. The Claims Are Not Indefinite Pursuant to 35 U.S.C. § 112, Second Paragraph

The Examiner rejected claims 1, 2, 6, 21, 22, 26, 27, and 31 as being indefinite under 35 U.S.C. § 112, second paragraph. The Office Action asserts that portions of the claim are "very vague and confusing with so many limitations beginning with 'wherein'" and that the claim language is "redundant." Applicant has amended claims 1, 2, 6, 21, 22, 26, 27, and 31 for clarification. Applicant requests removal of the rejections under 35 U.S.C. § 112, second paragraph.

Regarding claim 2, the Office Action further states:

> Claim 2, page 36, line 5 contains a conditional statement. It is unclear what happens "if the first data does not compare equally to the FSO transaction-related data contained in the fields of the first FSO..." Clarification of the claim language is respectfully requested.

Applicant respectfully submits that amended claim 2 is not unclear. Amended claim 2 recites in part: "access and read the first processing parameter if the first data compares equally to the FSO transaction-related data contained in fields of the first FSO transaction-related data that are identified by the selected plurality of field identifiers read from the first memory." Claim 2 is directed to a method in which a computer system is configured such that, if the recited condition is satisfied, the computer system will read and access a first processing parameter. Applicant respectfully submits that claim 2 need not address a state in which the recited condition is not satisfied.

Regarding claims 6, 31, and 78, the Office Action states:

> Claims 6, 31, and 78 are not sufficiently precise due to the combining of two separate statutory classes of invention in a single claim. The preamble of the claim refers to a method, but the body of the claim discusses the specifics of the system of a Financial Service Organization (FSO) (ex. displaying one or more key elements, and selecting one or more key element representations), and subsequently the claim then deals with the specifics of a method (the step ex. preparing a key definition from the one or more key elements, storing the key definition, configuring the key definition stored in a database for use in preparing a processing key value from a transaction related data) in a FSO computer system.

Applicant disagrees that claims 6, 31, and 78 are not sufficiently precise. As stated in M.P.E.P §2173.05(p): "There are many situations where claims are permissively drafted to include a reference to more than one statutory class of invention." A single claim that claims both an apparatus and the method steps of using the apparatus is indefinite under 35 U.S.C. § 112, second paragraph. *Ex parte Lyell*, 17 U.S.P.Q.2d 1548 (Bd. Pat. App. & Inter. 1990). The independent claim at issue in *Lyell* recited:

> 2. An automatic transmission tool in the form of a workstand and method for using same comprising:
> a support means,
> and [sic] internally splined sleeve affixed upright to said support means,
> a threaded adjustment bolt threadably engaged through a hole in the bottom of said support means and projecting upward through said support frame into said sleeve;
> and further comprising the steps of:
> 1. positioning the output end of an automatic transmission onto said upright sleeve,
> 2. removing the internal components of said automatic transmission from the casing of said transmission,
> 3. repairing and replacing said internal components back into said casing,
> 4. adjusting said internal components by means for fit and interference by means of adjusting said upwardly projecting adjustment bolt.
> *Lyell*, 17 U.S.P.Q.2d at 1548. (emphasis added)

In contrast to the claims at issue in *Lyell*, Applicant's claims 6, 31, and 78 do not claim both an apparatus and method, but claim methods only. In particular, amended claims 6, 31, and 78 recite in part: "A computer-implemented method comprising…." Applicant respectfully requests removal of the rejections of claims 6, 31, and 78 and the claims dependent thereon.

D. **The Claims Are Not Directed to Non-Statutory Subject Matter Pursuant to 35 U.S.C. § 101**

The Examiner rejected claims 6-20, 31, and 78 under 35 U.S.C. § 101 as being directed to non-statutory subject matter. The Office Action states:

> Applicant's claims mentioned above are intended to embrace or overlap two different statutory classes of invention as set forth in 35 USC 101. …A computer, a network or a machine must be in the preamble and the body of the claim to perform the method in order for the claim to comply with being technological and patentable.

In support of the rejections under 35 U.S.C. §101, the Office Action cites the *Lyell* case discussed in Section C above. As noted above, the individual claims at issue in *Lyell* claimed both an apparatus and a method. Specifically, the claim preamble recited: "An automatic transmission tool in the form of a workstand and method for using same comprising…." (emphasis added). In contrast, Applicant's claims 6, 31, and 78 are directed to one statutory class only, namely, to a process. Applicant therefore respectfully disagrees with the rejections of Applicant's claims under 35 U.S.C. §101. Nonetheless, to expedite prosecution of the application, Applicant has amended claims 6, 31, and 78 in accordance with the Examiner's suggestion to recite: "A computer-implemented method comprising…" (emphasis added). Applicant requests removal of the rejections under 35 U.S.C. § 101 of claims 6, 31, and 78 and the claims dependent thereon.

E. **The Claims Are Not Obvious over Zaiken in View of Kanai Pursuant To 35 U.S.C. § 103(a)**

The Examiner rejected claims 1-5 and 21-30 under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 5,907,848 to Zaiken et al. (hereinafter "Zaiken") in view of U.S. Patent No. 5,864,679 to Kanai (hereinafter "Kanai"). Applicant respectfully disagrees with these rejections.

To reject a claim as obvious, the Examiner has the burden of establishing a *prima facie* case of obviousness. *In re Warner et al.,* 379 F.2d 1011, 154 USPQ 173, 177-178 (CCPA 1967). To establish a *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 U.S.P.Q. 580 (C.C.P.A. 1974), MPEP § 2143.03. In addition, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

Amended claims 1, 21, and 26 describe combinations of features including: “wherein the computer system is configured to: read the selected plurality of field identifiers from the first memory in response to the computer system receiving a first FSO transaction-related data, and access and read a first processing parameter from the second memory using FSO transaction-related data contained in fields of the first FSO transaction-related data that are identified by the selected plurality of field identifiers read from the first memory.” Zaiken and Kanai, considered separately or in combination, do not appear to teach or suggest at least these features of claims 1, 21, and 26, in combination with the other features of the claims.

The Office Action asserts that the above-quoted features of claims 1, 21, and 26 are taught in Kanai. Applicant respectfully disagrees with this assertion. Neither the portions of Kanai cited in the Office Action nor other portions of Kanai appear to teach or suggest at least these features. Kanai discloses:

> A multiple processor transaction processing system having both a transaction routing unit for routing each transaction generated by the transaction source to one of the transaction processors, and a data arrangement unit for determining data arrangement of the data to be used in processing the transactions by the transaction processors.
> (Kanai, abstract).

Kanai states:

> Also, in this case, each transaction TR is given in a form shown in FIG. 10, which includes a header TR-1 indicating the type of transaction and a number of arguments TR-2-1 to TR-2-4 required in processing this transaction. For instance, the example of FIG. 10 shows the transaction called "WITHDRAW" which carries out the withdrawal of the deposit from the bank account, and which has four arguments including "ACCOUNT__ID" indicating the bank account number, "TELLER__ID" indicating the number assigned to the automatic teller machine, "BRANCH__ID" indicating the number assigned to the branch of the bank, and "AMOUNT" indicating an amount to be withdrawn from the deposit.
>
> The example of the routing table shown in FIG. 9 indicates that, for the

> transaction "WITHDRAW", the transaction routing is made by looking up the argument "ACCOUNT__ID" and the database "ACCOUNT". Namely, when the transaction "WITHDRAW" arrives at the transaction routing unit, the routing determination unit 4A looks up the routing table 4C to ascertain that this transaction "WITHDRAW" is to be routed according to the values of argument "ACCOUNT__ID" and the database "ACCOUNT". Then, the value of the argument "ACCOUNT__ID" in the transaction is taken out. Then, the transaction routing unit looks up the data arrangement table 4B to check whether there exists the data in the database "ACCOUNT" which has the whose value is the same as the value of the argument "ACCOUNT__ID" of the transaction or not. Then, the processing of the transaction is commanded to the transaction processor checked in this manner.
> (Kanai, column 15, lines 33-62)
>
> Kanai further states:
>
> In the transaction routing unit 101 of the system shown in FIG. 11, the transaction transmitted from the transaction source processor 102 is received by the transaction reception unit 103, which supplies the received transaction to the feature parameter extraction unit 104 and the transaction transmission unit 106. The feature parameter extraction unit 104 extracts the feature parameters from the supplied transaction, and supplies it to the transaction processor selection unit 105 as well as to the processing history information memory unit 108 in which it is stored as the feature parameters of this transaction. The transaction processor selection unit 105 compares the values of the feature parameters supplied from the feature parameter extraction unit 104 and the processing history information of the past recorded in the processing history information memory unit 108, to select the optimum transaction processor for processing the transaction, and notifies the selected transaction processor to the transaction transmission unit 106 as well as to the processing history information memory unit 108 in which it is stored as the transaction processor which processed this transaction. The transaction transmission unit 106 transmits the transaction supplied from the transaction reception unit 103 to the transaction processor 110 notified from the transaction processor selection unit 105, to complete the transaction routing processing.
> (Kanai, column 19, lines 14-38)

Kanai appears to disclose a multiple processor transaction processing system that routes transactions to one of multiple processors. A transaction routing unit looks up information in a routing table and a data arrangement table to check the transaction processor and related database. Values of "feature parameters" extracted from the transaction are compared to the processing history information of the past recorded in the processing history information memory to select the optimum transaction processor for processing the transaction. Kanai does not appear to teach or suggest a computer system configured to read a selected plurality of field identifiers from a first memory in response to the computer system receiving a first FSO transaction-related data, and to access and read a first processing parameter from a second memory using FSO transaction-related data contained in fields of the first FSO transaction-related data that are identified by the selected plurality of field identifiers.

Moreover, Applicant respectfully submits that the Examiner has not stated a *prima facie* case of obviousness for combining Zaiken and Kanai. The Office Action states:

> Zaiken failed to teach, wherein the computer system is configured to receive a first FSO transaction related data, wherein the computer system is configured to read the selected plurality of field identifiers from the first memory in response to the computer system receiving the first FSO transaction-related data, wherein the computer system is configured to access and read a first processing parameter from the second memory using FSO transaction related data contained in fields of the first FSO transaction-related data that are identified by the selected plurality of field identifiers read from the first memory, and wherein the computer system is configured to process the first FSO transaction-related data and the first processing parameter.

The Office Action nonetheless asserts that all of the above features missing from Zaiken are taught in Kanai. The Office Action states:

> It would have been obvious to one of having ordinary skill in the art at the time the invention was made to have a computer system configured to receive a first FSO transaction related-data, wherein the computer system is configured to read the selected plurality of field identifiers from the first memory in response to the computer system receiving the first FSO transaction-related data, wherein the computer system is configured to access and read a first processing parameter from the second memory using FSO transaction related data contained in fields of the first FSO transaction-related data that are identified by the selected plurality of field identifiers read from the first memory, and wherein the computer system is configured to process the first FSO transaction-related data and the first

> processing parameter and to modify in Zaiken in view of Zaiken's teachings of field identifiers and key values because such a modification would allow Zaiken to have a transaction processing system for executing some kind of processing for transactions received from transaction sources.

Applicant respectfully disagrees with the Office Action's assertions. Zaiken discloses:

> A method and system of providing external transaction protection for a database using the database log or journal. The method involves creating a set of transaction templates which define transactions, using the templates to determine whether each record or entry in the journal represent part of a transaction, and maintaining a set of index file indicating transactions in progress.... In the event the database is damaged, existing index files are used to determine which transactions did not complete before the database was damaged. The actions which were completed may be rolled back.
> (Zaiken, abstract).

Zaiken discloses a method and system of providing external transaction protection for a database. If the database is damaged, existing index files are used to determine which transactions did not complete before the database was damaged. There does not appear to be any reason to modify Zaiken to allow Zaiken to have a transaction processing system to execute processing of transactions. Moreover, the showing of a suggestion, teaching, or motivation to combine prior teachings "must be clear and particular . . .. Broad conclusory statements regarding the teaching of multiple references, standing alone, are not 'evidence'." *In re Dembiczak*, 175 F.3d 994, 50 U.S.P.Q.2d 1614 (Fed. Cir. 1999). The art must fairly teach or suggest to one to make the specific combination as claimed. Applicant respectfully submits that to "allow Zaiken to have a transaction processing system for executing some kind of transactions received from transaction sources" does not provide a motivation to combine Zaiken with Kanai to include the several features of claims 1 and 21 quoted above.

Applicant submits that, for at least the reasons discussed above, claims 1, 21, and 26 and the claims depending thereon are patentable over the cited art. Applicant therefore respectfully requests the removal of the 35 U.S.C. §103(a) rejections of these claims.

Moreover, Applicant submits that many of the claims dependent on claims 1, 21, and 26 are independently patentable. For example, claim 2 describes a combination of features including: "access and read the first processing parameter if the first data compares equally to the FSO transaction-related data contained in fields of the first FSO transaction-related data that are identified by the selected plurality of field identifiers read from the first memory." The cited art does not appear to teach or suggest at least these features of claim 2, in combination with the other features of the claim.

The Office Action apparently takes the position that the above-quoted features of claim 2 are taught in Zaiken. Applicant respectfully disagrees with the Office Action's position. Neither the portions of Zaiken cited in the Office Action nor other portions of Zaiken appear to teach or suggest at least these features. Zaiken states:

> The template ID specified by the user is then compared to the IDs for any existing templates, step 68. If the specified template ID does not match any existing template IDs, indicating that the template will be new, the data values retrieved from the log record are compared to any data values previously retrieved from other log records having the same job identifier data, step 70. If the current log record represents the first record retrieved from the log during a given LEARN session, then no other data values will have been previously retrieved and the data values are stored to be used for comparison to subsequently retrieved records. If other data values have been stored, the data values from the currently retrieved log record are compared to each of the previously retrieved data values, step 72. If no data values match, then the data values from the retrieved log record are stored, step 74, and the next log record is retrieved, step 62.
> (Zaiken, column 9, lines 26-41)
>
> If some of the data fields in the currently retrieved log record match the key value, the program retrieves the filename from the log record, step 88. The filename is compared to the filenames already represented in the template, step 90. If the filename from the log record does not match a filename in the template, the filename is added to the template, step 94, and the next record is retrieved, step 62. If the filename from the log record does match a filename already represented in the template, and if the user has specified that duplicate filenames are not

> allowed in a transaction (because all actions to be performed on a file would be performed in a single step and stored in a single record), step 92, this serves as an indication that the job has begun a new transaction, and the transaction is considered complete and the template is closed.
> (Zaiken, column 10, lines 19-34)

Zaiken appears to disclose comparing data values from the currently retrieved log record to previously retrieved data values during a "LEARNING" session to create new transaction templates (see column 8, line 65 to column 9, line 1). Zaiken appears to further disclose comparing a currently retrieved log record with a key value, and retrieving a filename from the log record. Zaiken does not appear to teach or suggest a computer system configured to access and read a first processing parameter if a first data compares equally to the FSO transaction-related data contained in fields of the first FSO transaction-related data that are identified by a selected plurality of field identifiers.

Claim 3 describes a combination of features: "preparing a first processing key value from data contained in fields of the first FSO transaction-related data that are identified by the selected plurality of field identifiers read from the first memory." Claim 1, from which claim 3 depends, describes in part: "selecting a plurality of displayed field identifiers." The cited art does not appear to teach or suggest at least the above-quoted features of claim 3, in combination with the other features of the claim.

The Office Action asserts that the above-quoted features of claim 3 are taught in Kanai. Applicant respectfully disagrees with this assertion. Kanai states:

> The transaction table 126 is in a form shown in FIG. 29 in which each entry has five fields for registering the type of transaction, the argument information, the weight information, the transaction processor information, and the processing history information pointer pointing to the processing history information in the processing history information memory unit 108 in correspondence.
>
> The type of transaction is given by a value indicating the type of transaction at the

> top of the transaction packet. In the example shown in FIG. 29. "proc1" and "proc2" are the values indicating the types of transaction, and actually the numerical values corresponding to these "proc1" and "proc2" are registered. The argument information registers the data types of the arguments of the transaction for each type of transaction. In the example shown in FIG. 29, the type of transaction "proc1" has three arguments, whose data types are "int" (integer type), "string" (character string type), and "float" (floating point type). Actually, an array of numerical values assigned to "int", "string", and "float" are registered.
>
> It is to be noted that the data types such as "int", "string", and "float" used in FIG. 29 are all in a form without any structure, but in a case of using a structural entity as indicated in FIG. 30A as an argument, the argument information is given, as shown in FIG. 30B, as a set of a numerical value for "struct" which indicates that it is a structural entity, a number of members constituting this structural entity such as "2", and numerical values for the data types of the members such as "int" and "char". Also, in a case of using an array as indicated in FIG. 30C, the argument information is given, as shown in FIG. 30D, as a set of a numerical value for "array" which indicates that it is an array, a number of elements of the array such as "10", and a numerical value for the data type of each element such as "int". It is also possible for such a structural entity or an array to have its constituent element given by a structural entity or an array.
> (Kanai, column 23, lines 22-65)

Kanai appears to disclose a transaction table in which each entry has five fields for registering information relating to a transaction. The type of transaction is given by a value indicating the type of transaction at the top of the transaction packet. Kanai does not appear to teach or suggest preparing a first processing key value from data contained in fields of a first FSO transaction-related data that are identified by a plurality of field identifiers selected from displayed field identifiers.

F. Additional Remarks

Based on the above, Applicant submits that all claims are in condition for allowance. Favorable reconsideration is respectfully requested.

If any extension of time is required, Applicant hereby requests the appropriate extension of time. If any fees are required, please charge those fees to Meyertons, Hood, Kivlin, Kowert & Goetzel, P.C. Deposit Account Number 50-1505/5053-31401/EBM.

Respectfully submitted,

Mark R. DeLuca
Reg. No. 44,649
Patent Agent for Applicant

MEYERTONS, HOOD, KIVLIN, KOWERT & GOETZEL, P.C.
P.O. BOX 398
AUSTIN, TEXAS 78767-0398
Ph. 512-853-8800 Fax 512-853-8801

Date: 10/28/04